



工业和信息化“十三五”
人才培养规划教材



STM32 嵌入式 技术 应用开发|全案例实践

Cases Practice of STM32 Embedded Technology Application and Development

苏李果 宋丽 ◎ 主编

张叶茂 ◎ 副主编



“嵌入式技术应用开发”技能大赛培训推荐用书。

从初学者角度出发，精选 **STM32 嵌入式技术** 应用开发 **企业实际案例**；

项目引领，任务驱动；**知识点与技能点** 同任务实施过程有机结合；

深入浅出讲解 **STM32 微控制器** 的 **基础知识、硬件平台、工作原理与开发技巧**。



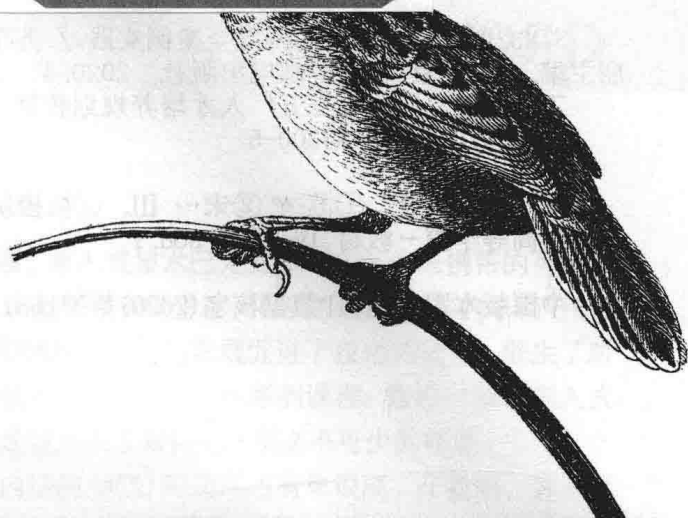
中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS



工业和信息化“十三五”
人才培养规划教材



STM32 嵌入式 技术

应用开发 | 全案例实践

Cases Practice of STM32 Embedded Technology Application and Development

苏李果 宋丽 ◎ 主编

张叶茂 ◎ 副主编

人民邮电出版社

北 京

图书在版编目 (C I P) 数据

STM32嵌入式技术应用开发全案例实践 / 苏李果, 宋丽主编. — 北京: 人民邮电出版社, 2020. 4
工业和信息化“十三五”人才培养规划教材
ISBN 978-7-115-53300-5

I. ①S… II. ①苏… ②宋… III. ①微控制器—系统开发—高等学校—教材 IV. ①TP368.1

中国版本图书馆CIP数据核字(2020)第031681号

内 容 提 要

本书主要介绍 STM32F4 系列微控制器在工程应用中所需的基础知识、硬件外设的工作原理以及编程开发的相关技术。本书由 5 个项目共 16 个任务构成, 介绍 STM32 的基础知识、基于 STM32F4 标准外设库的工程建立和 STM32F4 系统时钟的配置, 并结合可控 LED 流水灯的设计与实现、智能小车运动控制系统的设计与实现、环境参数监测和显示系统的设计与实现、多机通信系统的设计与实现等案例, 对 GPIO、外部中断、USART、ADC、定时器、SPI、I²C、SDIO、CAN 等常用硬件外设进行讲解。

本书在内容组织上采用了“项目引领、任务驱动”的模式。针对 MCU 常用的硬件外设, 设定了对应的应用场景作为案例, 并在讲解的过程中将所需的知识点和技能点同任务实施过程有机结合了起来。案例中还融入了全国职业院校技能大赛“嵌入式技术应用开发”赛项的考核点, 按照任务分析—知识链接—任务实施的路线对其展开了讨论, 十分有利于读者学习与实践。

本书可作为电子信息类和通信类专业的教学用书, 也可作为从事嵌入式产品设计、智能终端设备开发的工程技术人员的学习用书。

-
- ◆ 主 编 苏李果 宋 丽
 - 副 主 编 张叶茂
 - 责任编辑 祝智敏
 - 责任印制 王 郁 马振武
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
 - 邮编 100164 电子邮件 315@ptpress.com.cn
 - 网址 <http://www.ptpress.com.cn>
 - 三河市中晟雅豪印务有限公司印刷
 - ◆ 开本: 787×1092 1/16
 - 印张: 20 2020 年 4 月第 1 版
 - 字数: 481 千字 2020 年 4 月河北第 1 次印刷
-

定价: 62.00 元

读者服务热线: (010)81055256 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京东工商广登字 20170147 号

随着电子技术、计算机技术、通信技术的发展,嵌入式技术已无处不在。从随身携带的可穿戴智能设备,到智慧家庭中的远程抄表系统、智能洗衣机和智能音箱,再到智慧交通中的车辆导航、流量控制和信息监测等,各种创新应用及需求不断涌现。行业的发展促进了技术的进步、催生了对人才的需求,很多高校的电子信息技术专业都针对嵌入式技术开设了一系列课程。微控制器是嵌入式系统的核心,掌握一种微控制器的编程应用技术是嵌入式工程师的一项必不可少的技能。

在全球微控制器领域,基于 Arm Cortex-M 内核的 MCU 的市场占有率很高。在我国,意法半导体公司的 STM32 系列 MCU 几乎占据了 Cortex-M 内核微控制器领域的半壁江山。目前,国内针对 STM32 系列 MCU 的书籍以介绍理论知识居多,辅以简单的实验对理论加以验证,应用开发实践类型的书籍成为了广大智能电子产品设计爱好者和初学者迫切需要的资料。在这样的背景下,编者结合多年的教学与工程实践经验,编写了这本《STM32 嵌入式技术应用开发全案例实践》。

本书没有采用大而全的框架结构,而是基于意法半导体公司的高性能 STM32F4 系列 MCU,介绍了在工程实践中比较常用的外设硬件,并以具体的应用场景为教学案例,引导读者进行应用的开发与实践。本书的参考学时为 64 学时,建议老师采用理论实践一体化的教学模式,理论与实践的学时比例为 4:6,各项目的知识重点与学时建议见下表。

项目	任务	知识重点	学时建议
项目 1 走进 STM32 的世界	任务 1.1 STM32 学习八问 任务 1.2 STM32F4 标准外设库工程的建立 任务 1.3 STM32F4 系统时钟的配置	1. STM32 系列 MCU 相关的知识 2. 基于 STM32F4 标准外设库工程的建立步骤 3. STM32F4 系统时钟的配置方法	8 学时
项目 2 可控 LED 流水灯的设计与实现	任务 2.1 LED 流水灯的应用开发 任务 2.2 按键控制流水灯的应用开发 任务 2.3 串行通信控制流水灯的应用开发	1. STM32F4 的 GPIO 工作原理 2. STM32F4 的中断管理与配置 3. STM32F4 的 USART 外设的工作原理	10 学时
项目 3 智能小车运动控制系统的设计与实现	任务 3.1 智能小车循迹状态获取的应用开发 任务 3.2 智能小车供电监测模块的应用开发 任务 3.3 智能小车电机调速模块的应用开发 任务 3.4 智能小车电机测速模块的应用开发	1. STM32F4 的 ADC 外设的工作原理 2. STM32F4 的定时器基本定时、PWM 信号输出和硬件编码器功能 3. 智能小车电机调速与测速的实现原理	16 学时
项目 4 环境参数监测与显示系统的设计与实现	任务 4.1 环境温湿度监测的应用开发 任务 4.2 环境光照强度监测的应用开发 任务 4.3 环境参数持久化存储的应用开发 任务 4.4 具备交互功能的人机界面应用开发	1. 常用温湿度、光照强度传感器的工作原理 2. SD 存储卡的基本特性与读写原理 3. OLED 显示模块的工作原理 4. STM32F4 的 RTC 外设的工作原理 5. STM32F4 的 I ² C、SPI 和 SDIO 总线的工作原理 6. FatFs 文件系统的移植与使用方法	18 学时

续表

项目	任务	知识重点	学时建议
项目5 多机通信系统的设计与实现	任务 5.1 基于 RS-485 总线的多机通信应用开发	1. RS-485 和 CAN 标准	12 学时
	任务 5.2 基于 CAN 总线的多机通信应用开发	2. 自定义多机通信应用层协议的方法 3. STM32F4 的 bxCAN 外设的工作原理 4. STM32F4 标准外设库中与 CAN 相关的函数 API	

为了更好地服务广大师生，使本书更加适应当前教育信息化的需求，编写组制定了课程标准和教学计划，制作了精美的多媒体课件，编写了完整的教案，形成了一套完整的教学资源。读者可以登录人邮教育社区下载或联系编者获取教学资源。

本书由闽西职业技术学院的苏李果和宋丽担任主编，南宁职业技术学院的张叶茂担任副主编。苏李果编写了项目 2、项目 4 和项目 5，宋丽编写了项目 1 的任务 1.2、任务 1.3 和项目 3，张叶茂编写了项目 1 的任务 1.1。在本书的编写过程中，编者参考并引用了意法半导体公司提供的技术资料和应用笔记，听取了多方面的建议。本书得到了百科荣创（北京）科技发展有限公司的大力支持，该公司的黄文昌工程师为本书提供了教学案例，石浪总监对本书进行了审阅；本书介绍的智能小车设备由该公司提供，相关实验案例均可在其上运行。编者在此对他们表示衷心的感谢。同时感谢“全国高等院校计算机基础教育研究会计算机基础教育教学研究项目（2019-AFCEC-048）”课题组为本书编写所做的大量工作和所给予的支持。

由于编者水平有限，书中难免存在遗漏之处，恳请广大读者批评指正。读者可以将修改建议发至邮箱 Liguosun@qq.com，以与本书编者进行交流。

苏李果
2020 年 4 月

目录

CONTENTS

项目1 走进 STM32 的世界..... 1

任务 1.1 STM32 学习八问..... 2

1.1.1 什么是 STM32 2

1.1.2 什么是 Arm 4

1.1.3 用 STM32 能做什么 5

1.1.4 学习 STM32 必备的知识基础是什么 5

1.1.5 学习 STM32 需要哪些工具与平台,
有什么好的学习方法 6

1.1.6 如何搭建 STM32F4 系列微控制器的
最小系统 8

1.1.7 STM32 的软件开发模式有哪些 10

1.1.8 STM32F4 标准外设库的文件结构是
怎样的 12

任务 1.2 STM32F4 标准外设库 工程的建立 13

1.2.1 任务分析 13

1.2.2 知识链接 13

1.2.3 任务实施 16

任务 1.3 STM32F4 系统时钟的 配置 25

1.3.1 任务分析 25

1.3.2 知识链接 25

1.3.3 任务实施 28

项目2 可控 LED 流水灯的设计 与实现 33

任务 2.1 LED 流水灯的应用开发 34

2.1.1 任务分析 34

2.1.2 知识链接 34

2.1.3 任务实施 41

任务 2.2 按键控制流水灯的应用 开发 45

2.2.1 任务分析 45

2.2.2 知识链接 45

2.2.3 任务实施 53

任务 2.3 串行通信控制流水灯的 应用开发 62

2.3.1 任务分析 62

2.3.2 知识链接 62

2.3.3 任务实施 71

项目3 智能小车运动控制系统的 设计与实现 77

任务 3.1 智能小车循迹状态获取的 应用开发 78

3.1.1 任务分析 78

3.1.2 知识链接 79

3.1.3 任务实施 87

任务 3.2 智能小车供电监测模块的 应用开发 91

3.2.1 任务分析 91

3.2.2 知识链接 91

3.2.3 任务实施 98

任务 3.3 智能小车电机调速模块的 应用开发 104

3.3.1 任务分析 104

3.3.2 知识链接 105

3.3.3 任务实施 119

任务 3.4 智能小车电机测速模块的 应用开发 129

3.4.1 任务分析 129

3.4.2 知识链接 130

3.4.3 任务实施 137

项目4 环境参数监测与显示系统的 设计与实现 147

任务 4.1 环境温湿度监测的应用 开发 148

4.1.1 任务分析	148
4.1.2 知识链接	148
4.1.3 任务实施	151
任务 4.2 环境光照强度监测的 应用开发	156
4.2.1 任务分析	156
4.2.2 知识链接	157
4.2.3 任务实施	164
任务 4.3 环境参数持久化存储的 应用开发	173
4.3.1 任务分析	173
4.3.2 知识链接	175
4.3.3 任务实施	200
任务 4.4 具备交互功能的人机 界面应用开发	227
4.4.1 任务分析	227
4.4.2 知识链接	228
4.4.3 任务实施	241

项目 5 多机通信系统的设计与实现 257

任务 5.1 基于 RS-485 总线的多机 通信应用开发	258
5.1.1 任务分析	258
5.1.2 知识链接	258
5.1.3 任务实施	262
任务 5.2 基于 CAN 总线的多机 通信应用开发	275
5.2.1 任务分析	275
5.2.2 知识链接	276
5.2.3 任务实施	298

参考文献 314

Chapter 1

项目 1

走进 STM32 的世界

项目描述

本项目是嵌入式技术学习的先导篇，介绍 STM32 微控制器的基础知识，阐述基于 STM32F4 标准外设库的工程的建立与配置，并重点讲解 STM32 开发的必备知识——STM32 F4 系统时钟的配置。



学习目标

- (1) 了解嵌入式系统的基本概念、STM32 微控制器的主要特性、内部结构与最小系统组成；
- (2) 了解 Keil μ Vision5 集成开发环境的常用菜单功能；
- (3) 了解 STM32 微控制器的典型应用、开发模式和学习方法；
- (4) 熟练掌握 STM32F4 系列微控制器的时钟系统构成；
- (5) 能正确安装并使用 Keil μ Vision5 集成开发环境；
- (6) 会建立并配置基于 STM32F4 标准外设库的工程；
- (7) 能根据应用需求进行 STM32F4 系统时钟的配置。



任务 1.1 STM32 学习八问

1.1.1 什么是 STM32

1. STM32 概述

STM32 微控制器是意法半导体 (ST Microelectronics, ST) 有限公司出品的一系列微控制器 (Micro Controller Unit, MCU) 的统称。

意法半导体有限公司于 1987 年 6 月成立, 由意大利的 SGS 微电子公司和法国的 Thomson 半导体公司合并而成 (以下简称 ST 公司), 是世界上最大的半导体公司之一。

STM32 微控制器基于 Arm Cortex®-M0、M0+、M3、M4 和 M7 内核, 这些内核是专门为高性能、低成本和低功耗的嵌入式应用设计的。STM32 微控制器按内核架构可以分为以下产品系列。

- 通用微处理器产品系列: STM32MP1。
- 高性能产品系列: STM32F2、STM32F4、STM32F7、STM32H7。
- 主流产品系列: STM32F0、STM32F1、STM32F3。
- 超低功耗产品系列: STM32L0、STM32L1、STM32L4、STM32L4+。
- 无线系列: STM32WB。

图 1-1-1 展示了 STM32 微控制器的产品家族。

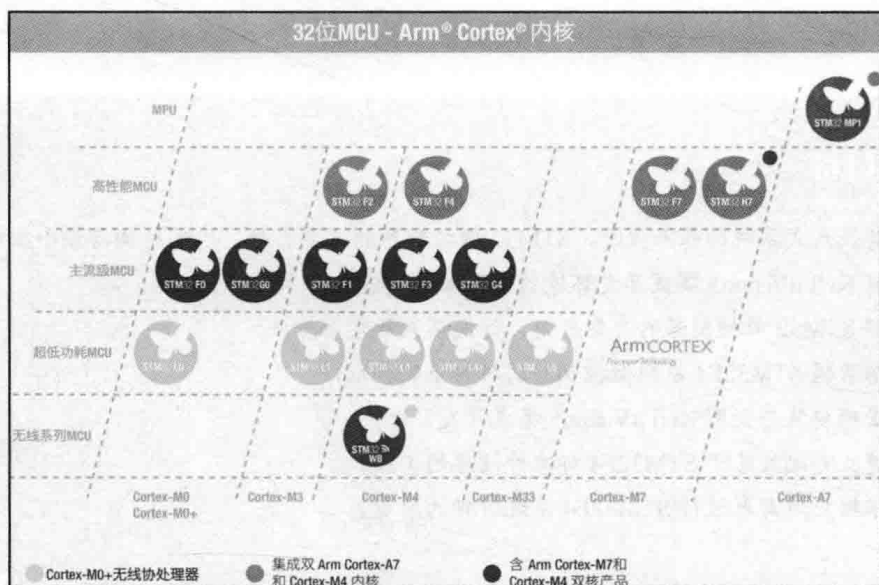


图 1-1-1 STM32 微控制器的产品家族

2. STM32 微控制器的命名规则

各个型号的 STM32 微控制器在封装形式、引脚数量、静态随机存储器 (SRAM) 和闪存的大小、最高工作频率 (影响产品的性能) 等方面有所不同, 开发人员可根据应用需求选择最合适的 STM32 微控制器来完成项目设计。STM32 微控制器型号的各部分含义介绍如图 1-1-2

所示。

STM32	F	051	R	8	T	6	X	XX
家族 STM32 32位MCU / MPU	特定功能 (3位数字) (依据产品系列 非详细列表) STM32x... 051 入门级 103 STM32基础型 303 103升级版, 带DSP 和模拟外设 407 高性能, 带DSP和MPU 152 超低功耗	闪存容量(Kbytes) 0 1 1 2 2 4 3 8 4 16 5 24 6 32 7 48 8 64 9 72 A 96 B 128 C 192 D 256 E 384 F 512 G 768 H 1024 I 1536 J 2048	封装 B Plastic DIP* D Ceramic DIP* G Ceramic QFP H LFBGA / TFBGA I UFBGA Pitch 0.5** J UFBGA Pitch 0.8** K UFBGA Pitch 0.65** M Plastic SO P TSSOP Q Plastic QFP T QFP U UFQFPN V VFQFPN Y WLCSP ** Dual-in-Line封装 ** 仅针对全新产品系列 现有产品系列请使用H	固件版权 U Universal 不用于生产 (样品和工具) V MP3解码器 W MP3编解码器 J 0.80 mm D IS21 JAVA	选项 xxx Fastrom code or xTR Tape and Reel Dxx No RTC (STM8L) Dxx BOR OFF with Special bonding + Boot standard Dxx BOR OFF with Boot I2CS (Special) Sxx BOR OFF Ixx BOR ON No Letter BOR ON + Boot standard or Yxx Die rev (Y)			
产品类别 A 汽车级 F 基础型 L 超低功耗 S 标准型 WB 无线产品 H 高性能 G 主流型	引脚数 D 14引脚 Y 20引脚(STM8) F 20引脚(STM32) E 24 & 25引脚 G 28引脚 K 32引脚 T 36引脚 H 40引脚 S 44引脚 C 48 & 49引脚 U 63引脚 R 64 & 66引脚 J 72引脚 M 80引脚 O 90引脚 V 100引脚 Q 132引脚 Z 144引脚 A 169引脚 I 176 & 201 (176+25)引脚 B 208引脚 N 216引脚 X 256引脚 汽车级 8 48 9 64 A 80	温度范围(°C) 6和A -40到+85 7和B -40到+105 3和C -40到+125 D -40到+150						

图1-1-2 STM32微控制器型号各组成部分含义介绍

下面以一个具体的 STM32 微控制器型号（STM32F407ZGT6）为例来介绍型号中各部分的含义，如表 1-1-1 所示。

表 1-1-1 STM32 微控制器型号 STM32F407ZGT6 的各部分含义介绍

序号	组成部分	具体含义
1	STM32	代表 ST 公司出品的基于 Arm Cortex®-M 内核的 32 位微控制器
2	F	代表“基础型”产品类别
3	407	代表“高性能”产品系列
4	Z	代表 MCU 的引脚数，如 T 代表 36 脚，C 代表 48 脚，R 代表 64 脚，V 代表 100 脚，Z 代表 144 脚，I 代表 176 脚等
5	G	代表 MCU 的内存容量，如 6 代表 32KB，8 代表 64KB，B 代表 128KB，C 代表 256KB，D 代表 384KB，E 代表 512KB，G 代表 1MB
6	T	代表 MCU 的封装，如 H 代表 BGA 封装，T 代表 QFP 封装，U 代表 UFQFPN 封装
7	6	代表 MCU 的工作温度范围，如 6 和 A 均代表-40~85℃，7 和 B 均代表-40~105℃

3. STM32F4 系列微控制器的系统架构

STM32F4 系列微控制器在片上集成了各种功能部件，各部件之间通过总线相连。这些功能部件包括内核（Core）、系统时钟发生器、复位电路、程序存储器、数据存储器、中断控制器、调试接口以及各种外设等。

STM32F4 系列微控制器的片上外设：通用输入/输出（GPIO）端口、定时器（Timer）、模数转换器（ADC）、数模转换器（DAC）、通用同步/异步收发器（USART）、安全数字输入/输出（SDIO）接口、串行外设接口（SPI）、内部集成电路（IIC）接口、控制器区域网络（CAN）总线等。

图 1-1-3 展示了 STM32F4 系列微控制器的系统架构。

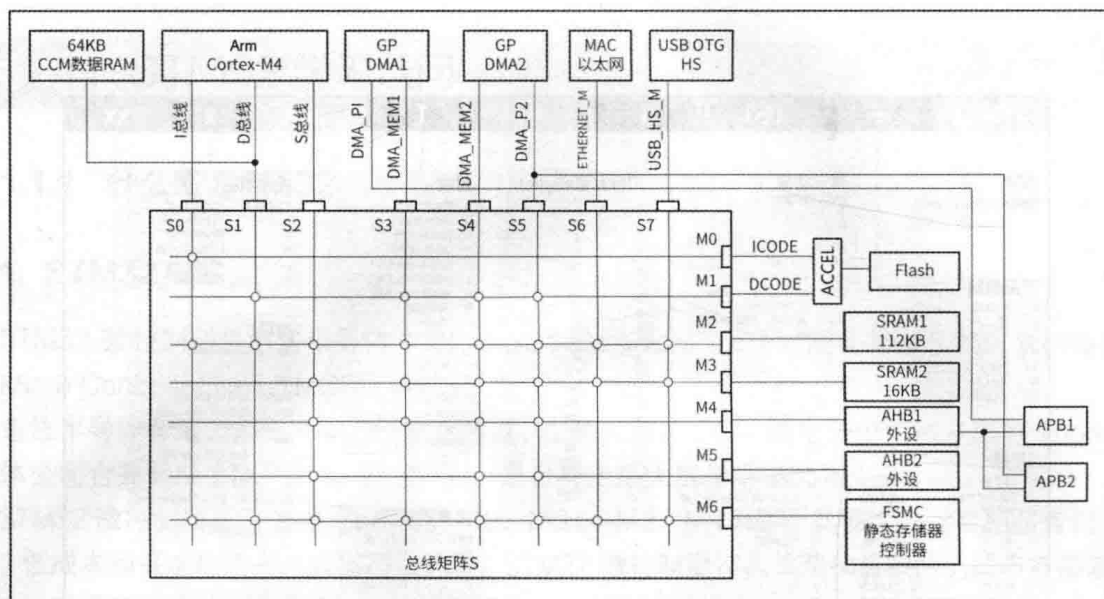


图1-1-3 STM32F4系列微控制器的系统架构

从图 1-1-3 中可以看到，主系统由 32 位多层先进的高性能总线（AHB）矩阵构成。借助 AHB 矩阵，MCU 可以实现主控总线到被控总线的访问。这样，即使多个高速外设同时运行，系统也可以实现并发访问和高效运行。

1.1.2 什么是 Arm

Arm(Advanced RISC Machines)公司是微处理器行业中的一家知名企业，它设计了大量高性能、低能耗且廉价的微处理器。Arm 公司的产品适用于多个领域，如嵌入式控制、消费/教育类多媒体、数字信号处理和移动式应用等。目前全世界超过 95% 的智能手机和便携式计算机的微处理器都采用 Arm 架构。

1990 年 11 月，Arm 公司在英国剑桥正式成立，专门从事基于精简指令集计算机（Reduced Instruction Set Computer, RISC）技术的芯片设计开发工作。作为一家知识产权供应商，Arm 公司并不直接生产芯片，只出售芯片设计技术的授权，而由合作公司生产具体的芯片。世界各大半导体生产商从 Arm 公司购买其设计的微处理器核心，根据不同的应用领域加入适当的外围电路，生产出各具特色的基于 Arm 内核的微处理器芯片。目前全世界有几十家半导体公司与 Arm

公司签订了硬件技术使用许可协议，其中包括 Intel、IBM、三星半导体、NEC、SONY、飞利浦和美国国家半导体等大公司。Arm 公司软件系统的合作伙伴包括微软、SUN 和 MRI 等知名公司。图 1-1-4 展示了 Arm 公司与合作伙伴的关系。

综上所述，“Arm”既代表一家公司（从事基于 RISC 技术的芯片设计工作，只出售芯片设计技术的授权）的名称，又代表一种技术（微处理器核心设计），还代表一类微处理器（由半导体公司生产的基于 Arm 架构的微处理器）。

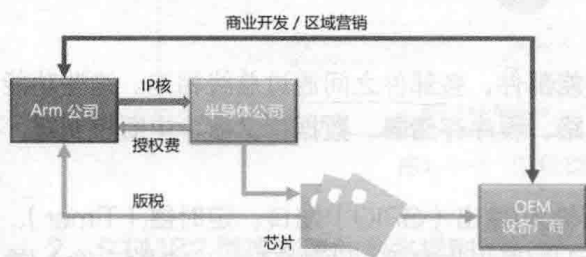


图1-1-4 Arm公司与合作伙伴的关系

Arm 处理器经过多年的发展,产生了 Arm v1、v2、v3、v4、v5、v6、v7、v8 等不同版本的内核架构。自 Arm v7 架构开始,内核命名将 Cortex 作为前缀,分为 Cortex-A、Cortex-R 和 Cortex-M 三大系列。三大系列分工明确: Cortex-A 系列内核面向尖端的基于虚拟内存的操作系统和用户应用,如移动领域的微处理器(MPU); Cortex-R 系列内核面向对实时性要求较高的应用; Cortex-M 系列内核主要用于微控制器。STM32 系列微控制器就是基于 Arm v7 架构中的 Cortex-M 系列内核设计出来的。

1.1.3 用 STM32 能做什么

随着电子技术、计算机技术、通信技术的发展,嵌入式技术已无处不在。从随身携带的可穿戴智能设备到智慧家庭中的远程抄表系统、智能洗衣机和智能音箱,再到智慧交通中的车辆导航、流量控制和信息监测等,各种创新应用及需求不断涌现。

电子产品得以快速更新、淘汰,其组成部分中最基础的底层架构芯片——微控制器(MCU)功不可没。目前 MCU 已成为电子产品及行业应用解决方案中不可或缺的一部分。

ST 公司在 2007 年发布了首款搭载 Arm Cortex-M3 内核的 32 位 MCU。在之后的十余年时间里,STM32 产品线又相继加入了基于 Arm Cortex-M0、Cortex-M4 和 Cortex-M7 内核的产品,产品线覆盖了通用型、低成本、超低功耗、高性能低功耗以及甚高性能等类型。正是由于 STM32 系列微控制器拥有完整的产品线和简单易用的应用开发生态系统,越来越多的领域的电子产品才将 STM32 系列微控制器作为主流的解决方案,涵盖智能硬件、智能家居、智慧城市、智慧工业、智能驾驶等领域。图 1-1-5 是生活中的一些常见的使用 STM32 系列微控制器的电子产品。

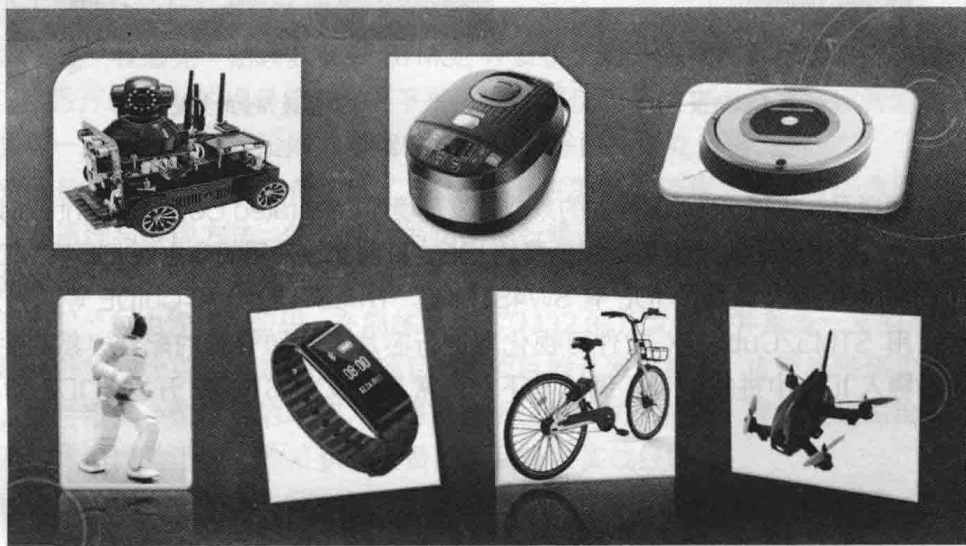


图1-1-5 使用STM32微控制器的电子产品

1.1.4 学习 STM32 必备的知识基础是什么

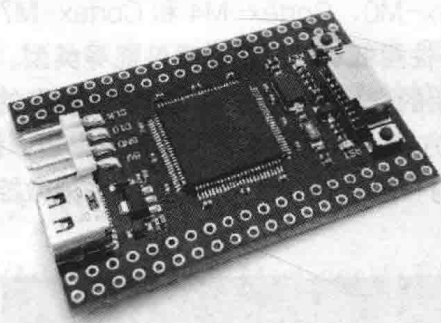
STM32 系列微控制器应用开发使用的编程语言主要是 C 语言。在应用开发的过程中,用户还经常需要阅读电路原理图、集成电路芯片的数据手册(Datasheet)等专业资料。因此在学习 STM32 前,应先学习“C 语言程序设计”“模拟电子电路分析与应用”“数字电子电路分析与应

用”等课程。上述课程可为学习 STM32 奠定模拟和数字电子电路分析设计的原理与方法、基本的 C 语言程序设计语法与规范等知识基础。

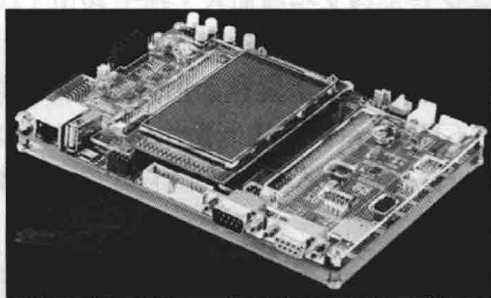
1.1.5 学习 STM32 需要哪些工具与平台，有什么好的学习方法

1. 学习 STM32 所需的工具与平台

在开始 STM32 学习之前，我们需要先挑选一块合适的 STM32 开发板。初学者不应盲目地追求开发板的功能，应以够用为原则，重点关注开发板配套的学习资料与视频是否详细、全面。目前市面上可供选购的 STM32 开发板主要有两种：最小系统板和外设齐全的开发板，分别如图 1-1-6 (a) 和图 1-1-6 (b) 所示。上述两种开发板各有优缺点：从价格上来说，最小系统板比外设齐全的开发板便宜；从提升硬件电路的构建能力来说，在使用最小系统板进行学习时，需要自行搭建外设的应用电路，这有助于学习者更好地理解外设电路的原理，并能够提高其电路板设计与制作的能力；从使用的便利性来说，外设齐全的开发板具有绝对的优势，学习者使用这种开发板可以方便地完成芯片性能的测试、程序功能的验证以及想法创意的快速应用。



(a) 最小系统板



(b) 外设齐全的开发板

图 1-1-6 STM32 开发板

ST 公司官网显示，支持 STM32 开发的集成开发环境 (Integrated Development Environment, IDE) 有 20 余种，其中包括商业版 IDE 和免费的 IDE。目前比较常用的商业版 IDE 有 MDK-ARM-STM32 和 IAR-EWARM，免费的 IDE 有 SW4STM32、TrueSTUDIO 和 CoIDE 等。另外，ST 公司官方推荐使用 STM32CubeMX 软件可视化地进行芯片资源和管脚的配置，然后生成项目的源程序，最后导入 IDE 中进行编译、调试与下载。常见的支持 STM32 开发的 IDE 如图 1-1-7 所示。



图 1-1-7 常见的支持 STM32 开发的 IDE

2. 学习STM32的方法

如果想在短时间内上手STM32微控制器的开发(即“入门STM32”),初学者需要采用科学的学习方法,制定一个完善的学习计划并严格按照计划实施。

首先,不要把时间过多地花在犹豫上。人们在学习一项新技术前,可能都会经历“犹豫期”。他们会先查阅资料,了解新技术并解决心中的一些疑问,如这个技术难不难、需要什么基础、适合不适合自己等。适当的前期调研是必要的,但过多地思索却没有实际行动是永远不可能入门的。因此,如果你对STM32开发感兴趣,那么请不要过多地犹豫,根据后面介绍的方法按部就班地开始学习吧。

然后,准备好必需的STM32开发板、开发软件和学习文档。STM32开发板与开发软件的选择可参考前文,此外,我们还需要准备以下几个必备的学习文档:一是《STM32F4xx中文参考手册》,它介绍了STM32F4系列微控制器各种外设的工作原理;二是《STM32标准外设库使用手册》,它介绍了STM32标准外设库函数API的调用方法和使用实例;三是某个型号的STM32微控制器的Datasheet,该文档有助于我们了解MCU的电气性能、管脚分布与外设功能;四是开发板生产厂商编制的产品使用手册。

最后,一个很重要的环节是制定学习计划。

(1) 快速浏览一遍文档

STM32开发涉及的文档内容很多,没必要也无法一次性全部看完,但STM32开发的通用基础部分必须得看,如存储器和总线架构、电源控制、备份寄存器、复位和时钟控制、GPIO引脚及其功能复用、中断等。具体某个外设的工作原理与使用方法在用到时再仔细阅读即可。

(2) 制定分阶段目标

STM32的学习过程可以分为以下3个阶段。

第一阶段是“找感觉”阶段。拿到STM32开发板之后,先把厂家配套的开发板使用手册浏览一遍,以熟悉开发板上的硬件组成。接下来可以按照使用手册中与开发板配套的测试例程的操作步骤,操作一遍开发板。本阶段的学习能让你找到感觉并建立自信。

第二阶段是“模仿”阶段。在了解了STM32开发的基本流程之后,可以选取一些例程,详细分析其工作原理与实现方法,并对例程的功能进行修改,以达到不同的运行效果。本阶段的学习能让你获得成就感。

第三阶段是“自由发挥”阶段。在熟练掌握STM32的开发流程并具备一定的开发经验之后,你可以选取并开发一个综合性较强的小项目。在项目的开发过程中,应严格按照实际的项目开发流程实施,不可遗漏一些重要的环节,如需求分析、系统功能描述、程序流程图绘制与软件文档编写等。通过本阶段的学习,你将积累宝贵的项目开发实战经验。

经过以上3个阶段的学习,你就可入门STM32微控制器的开发!

3. 经验之谈

(1) 遇到问题怎么办

先谈谈对待问题的态度:遇到问题时不要抱怨。俗话说得好:出现问题是给你学习的机会。因此解决问题的过程可以促进能力的提升。我们应摆正态度,正视问题。

另外,遇到问题时不要马上四处求助。当问题刚出现时,你可能还无法看清问题的全貌,也无法用最合适的语言将问题表述清楚以使别人理解并做出回答。因此在遇到问题时,应先查资料,

自己尝试解决问题。

如果经过一番分析,你仍然无法解决问题且必须向他人请教时,应注意提问的艺术,即不问不具体的问题。如“FreeRTOS 如何移植?”这个问题,需要非常大的篇幅才能阐述清楚,因此类似的问题是需要避免的。换个角度,如果我们将项目实施过程中遇到的具体问题提出来并向技术“大师”请教,有经验的人可能几句话就可以解答,因此这才是正确的提问方式。

(2) 学习过程中注意总结经验

在 STM32 的学习过程中,可能会遇到很多问题,你通过自己的努力解决了这些问题,并获得了成就感,但是过了一段时间你可能会忘记当时的解决方法与步骤。因此在学习过程中,花一点时间进行经验总结是非常有必要的。如果我们能将总结文档发布到博客中进行分享,则可为遇到相同问题的人提供帮助,何乐而不为呢?

1.1.6 如何搭建 STM32F4 系列微控制器的最小系统

一般来说,STM32F4xx 微控制器的最小系统应包含以下 6 个部分。

1. STM32F4xx 微控制器

STM32F407ZGT6、STM32F405RGT6 等型号的微控制器均属于 STM32F4xx 微控制器。

2. 电源

最小系统板的电源可采用多种不同的芯片组合方案。如果最小系统板采用 12V 直流电压适配器供电,则电源可采用“LM2596+ASM1117”的芯片组合方案,其中 LM2596 芯片将 12V 直流电压转换为 4V,ASM1117 芯片将 4V 直流电压转换为 3.3V 后为微控制器供电。电源的电路原理图如图 1-1-8 所示。

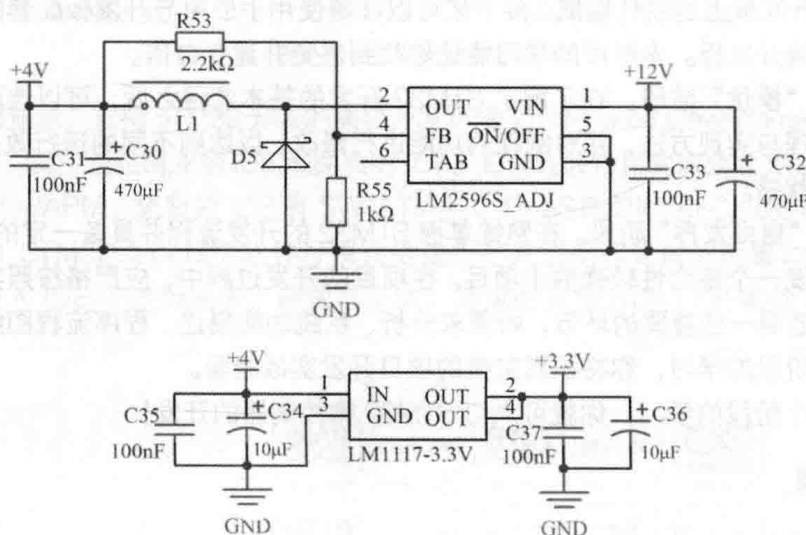


图1-1-8 电源部分的电路原理图

3. 外部晶振

STM32F4 系列微控制器的最小系统板需要安装两个外部晶振: 高频晶振(频率为 8MHz 或

者 25MHz) 和低频晶振 (频率为 32.768kHz)。外部晶振的电路原理图如图 1-1-9 所示。

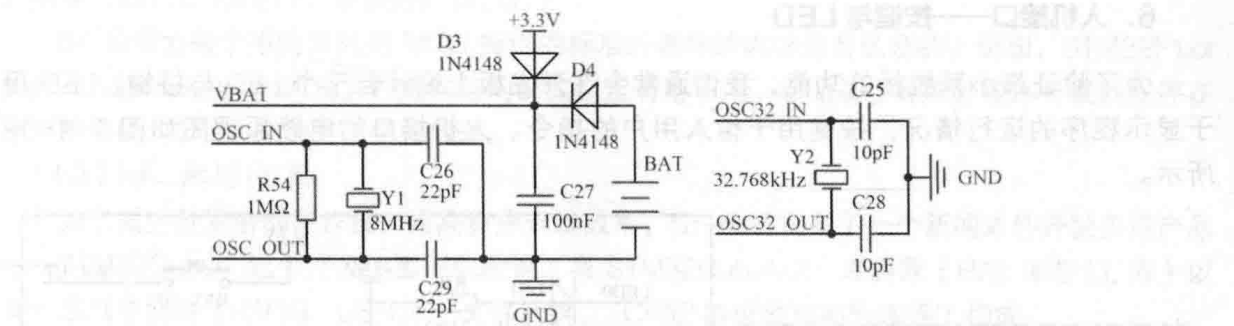


图1-1-9 外部晶振的电路原理图

4. JTAG 仿真调试接口

用户通过 JTAG 仿真调试接口将 STM32 开发板与 J-Link 或 ST-Link 仿真器相连，然后通过集成开发环境进行程序下载或完成在线调试。JTAG 仿真调试接口的电路原理图如图 1-1-10 所示。

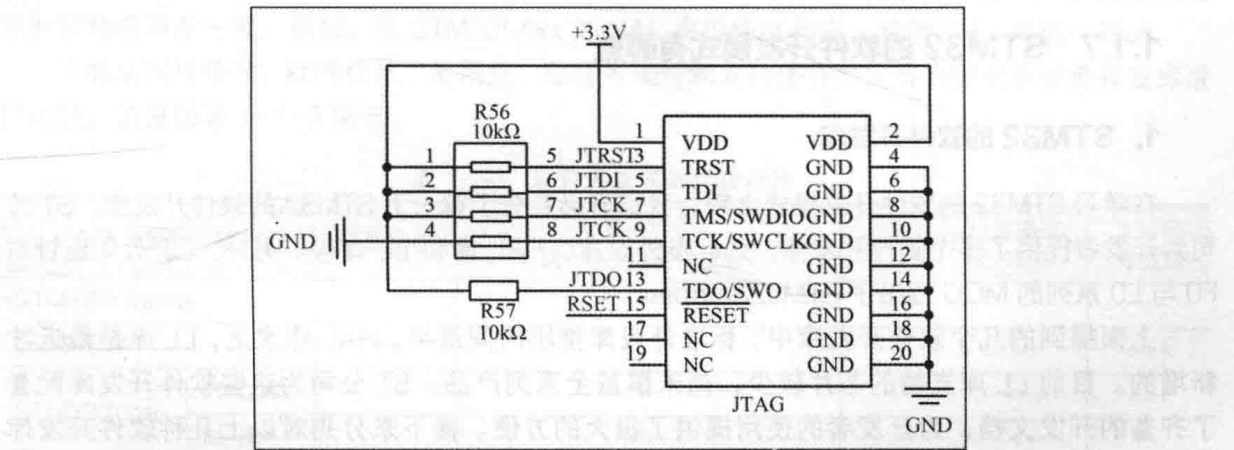


图1-1-10 JTAG仿真调试接口的电路原理图

5. 启动模式配置电路

STM32 微控制器有 3 种启动模式，它们对应的存储介质都是芯片内置的，分别是主 Flash 存储器、系统存储器和内置 SRAM。用户通过配置两个启动模式选择引脚（BOOT0 和 BOOT1）可选择从哪个存储介质启动。启动模式配置方法见表 1-1-2。

表 1-1-2 启动模式配置方法

启动模式选择引脚		存储介质	说明
BOOT1	BOOT0		
X	0	主 Flash 存储器	这是正常的启动模式
0	1	系统存储器	这种启动模式一般用于从串口下载程序
1	1	内置 SRAM	这种启动模式一般用于调试

启动模式配置电路的原理图如图 1-1-11 所示。

6. 人机接口——按键与 LED

为了验证最小系统板的功能，我们通常会在开发板上设计若干个 LED 与按键。LED 用于显示程序的运行情况，按键用于输入用户的指令。人机接口的电路原理图如图 1-1-12 所示。

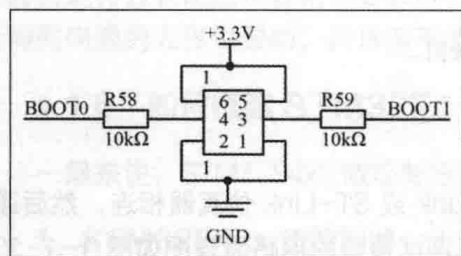


图1-1-11 启动模式配置电路的原理图

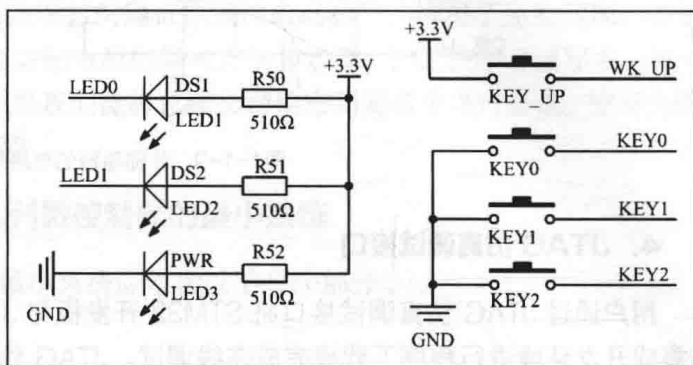


图1-1-12 人机接口的电路原理图

1.1.7 STM32 的软件开发模式有哪些

1. STM32 的软件开发库

在学习 STM32 的软件开发模式之前，我们有必要先了解一下 STM32 的软件开发库。ST 公司为开发者提供了多个软件开发库，如标准外设库、HAL 库和 LL 库等。另外，ST 公司还针对 F0 与 L0 系列的 MCU 推出了 STM32Snippets。

上面提到的几个软件开发库中，标准外设库推出时间最早，HAL 库次之，LL 库是最近才新增的。目前 LL 库支持的芯片较少，尚未覆盖全系列产品。ST 公司为这些软件开发库配套了齐备的开发文档，为开发者的使用提供了极大的方便。接下来分别对以上几种软件开发库进行介绍。

(1) STM32Snippets

STM32Snippets 是 ST 公司推出的提供高度优化且立即可用功能的寄存器级代码段集合，以最大限度地发挥 STM32 微控制器应用设计的性能和能效的软件开发库。寄存器级的编程虽然可降低内存占用率，节省宝贵的处理器时钟周期，降低电源电流消耗，但通常需要开发者花费很多时间和精力研究产品手册。另外，这种开发模式的缺点是代码在不同系列的 STM32 微控制器之间没有可移植性。

(2) 标准外设库

标准外设库 (Standard Peripherals Library) 是对 STM32 微控制器进行了完整封装的库，它包括了 STM32 微控制器所有外设的驱动描述和应用实例，为开发者访问底层硬件提供了一个中间函数 API。通过标准外设库，开发者无须深入掌握底层硬件的细节就可以轻松驱动外设，快速部署应用。因此，使用标准外设库可以减少开发者驱动片上外设的编程工作量，降低时间成本。